CHAPTER

# 6

# DESIGNING A DFS STRATEGY

**After reading this chapter and completing the exercises, you will be able to:**

- Describe the features, terminology, processes, and network activities of Dfs
- Understand the roles of functionality, security, availability, and performance in a Dfs design

Not long ago, most business networks were departmental LANs created to allow a small community of users to access shared resources. Those were the days when the old 80/20 rule of network usage was easy to apply, because 80% or more of the traffic was confined to the local network, while only 20% or less was needed to cross routers to access enterprise-wide resources.

In those "good old days," Microsoft's Server Message Block file-sharing protocol and supporting services seemed to be all we needed to handle the traffic on the network. Administrators only needed to create shares on individual servers and organize into those shares the resources that users required.

Ah, but everything grows, and the volume of files on networks was no exception. We had more and more resources being shared on more and more servers *and* we were switching to distributed computing in the enterprise networks. This further exacerbated file access and file distribution problems.

How can administrators manage all these resources, and how can we keep users up to date on the changes we are making in the locations of data and applications? One answer, depending on the requirements, is the Windows 2000 **Distributed File System (Dfs)**. In this chapter, you will examine the features of Dfs and how to plan for a Dfs implementation.

## Dfs—What You Need to Know Before You Start

Dfs is an enterprise-wide file management system. It allows administrators to hide the complexity of shared resource distribution from end users. With Dfs, users simply see a share on a server with a folder hierarchy beneath it. It does not look any different to the user than an ordinary share. In reality, however, the server is hosting a Dfs root folder along with the links describing the actual network location of the resources that users need to access. The Dfs root folder has the honor of being at the top of the namespace—all folders beneath it are contained within one Dfs namespace. It may not be at the top level (that is, the root) of the disk volume on which it resides, but it certainly defines the top of the Dfs namespace.


As you progress in your studies of network design, you'll come to appreciate the reuse of words. For instance, the word "root" is one we never seem to get tired of using—with a different meaning each time. In addition to the Dfs root, we use "root" to refer to the top-level folder on a disk volume. Actually, in the pre-GUI days of DOS, we referred to the top-level directory on a disk volume as a root directory. We still refer to the top-level folders that we see in REGEDIT.EXE as root keys or root trees. All this to say, our industry is quite loose with the word "root."


When Microsoft introduced the Dfs for Windows NT 4.0, they provided both the server and the client side for Windows NT 4.0 Server but just the client side for Windows 98 and Windows NT 4.0 Workstation. At this writing, however, Dfs 4.1 for Windows NT and Windows 9*x* can be downloaded from Microsoft's site. Make a habit of frequently checking out the Web pages for Windows NT (*www.microsoft.com/ntserver*) and Windows 2000 (*www.microsoft.com/windows2000*) for the latest updates.

Dfs is ideal for making file and folder resources available on member servers. A member server in Windows NT and Windows 2000 is a server that is a member of a domain but does not play the role of domain controller. Historically, member servers have provided services such as file and print services, WINS servers, DHCP servers, DNS servers, and application servers. The file- and print-sharing role has generally been departmental, meaning that each department or other functional group managed their own file systems without much regard for the needs of other departments that might require access to common files.

Using Windows 2000 and Dfs, administrators can take advantage of new features of Dfs and have identical links across the enterprise. In addition, Dfs is an ideal solution for making file resources available for a Web server. In fact, with Dfs, you can allocate a portion of the Dfs logical namespace to be published by Internet Information Services (IIS).

What is more important—at least from an administrator's point of view—is the fact that Dfs allows administrators to create for the user a simplified logical view of what can actually be a very complicated distribution of file shares on several servers. For example, when the user connects to a server that has been designated as a Dfs root server and views shares that appear to be on that server, the reality is that what the user sees as shares on the root server are Dfs links. These are simply pointers to the actual shares residing on other servers.

As an example of this concept, Figure 6-1 shows what a user would see when viewing a Dfs root server named Danville, which has links to shares on another computer. The Dfs root share (Human Resources) looks like an ordinary share, and the links (employee, recruit, and training) look like file folders directly below the share on the same server. Again, in reality, the links are only references to the actual shares. In this case, the links actually point to shares on the server Carmel, but there is no indication of this reality to the user.



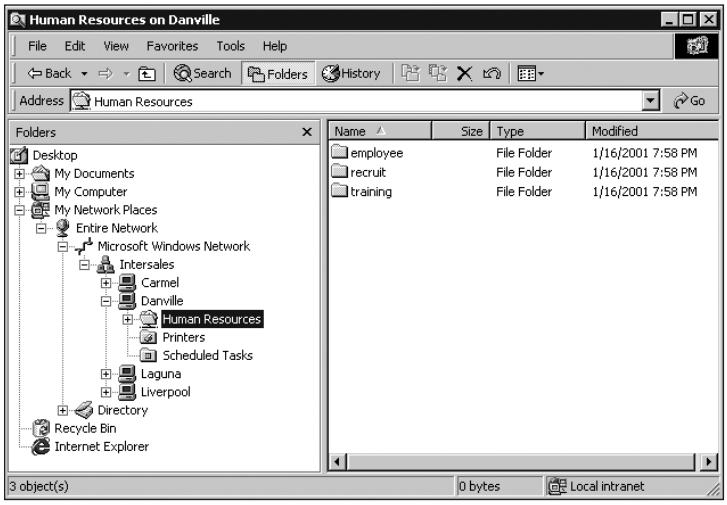


**Figure 6-1** The user's view of Dfs

To fully appreciate the scope of the power of Dfs, you need to know its features, the terminology that networking professionals use when working with it, and the processes and activities associated with the product itself. We will review all these next.

## Dfs Features and Benefits

When viewing any new product, it's best to start with the features and benefits. We'll do the same here because we're pretty pleased with the way Microsoft has enhanced Dfs for availability and fault tolerance in Windows 2000. Table 6-1 documents the features and benefits of Dfs in Windows 2000.

**Table 6-1** Windows 2000 Dfs: Features and Benefits

| Feature | Description | Benefit |
|---|---|---|
| Unified namespace | Folders containing file resources can exist in a variety of locations on many servers on the network, but appear to exist as a single hierarchy of folders on the same volume on one server, the Dfs root server. | Administrators can provide users with a single drive mapping that may actually involve hundreds of shared folders. This can be customized for individual users and for groups. |
| Name transparency | Use of the logical namespace of Dfs hides where things really are on the network. Users can continue to use the same root server, after administrators move the data they are accessing. Administrators only have to modify the locations in the logical namespace, using the Dfs console. | Administrators can move shared folders to different servers without having to modify the users' behavior for accessing the data in the new locations. |
| Flexible storage management | Administrators can take individual shared folders offline and not affect the remainder of the namespace. Servers can be removed and replaced by simply adding the path of the new server to a link in the namespace. | The management of the physical network storage is removed from the logical view the user sees. Administrators can easily extend the Dfs namespace to include additional disk storage. |
| Graphical administration | Administrators can administer each root using the Dfs console snap-in to the Microsoft Management Console. | Once an administrator has mastered working with any tool using the Microsoft Management Console, he or she can apply the same skills to using the Dfs console. |
| Load sharing | Load sharing is achieved when replicas of roots or shares are hosted on multiple computers. When a Dfs client requests connection to a replicated share in the Dfs namespace, the client randomly selects one of the replicas. | The file access load is distributed across multiple servers, balancing loads and improving response time. |
| Availability | Replica sets can be created under the same logical Dfs name from roots and shares that are hosted on two or more servers. When one copy becomes unavailable, the Dfs client automatically selects another. | The use of replicas in Dfs means that data will usually be available even if there is a failure of a server or drive. |
| Security integration | Dfs does not maintain separate access control lists. Rather, access is controlled by the permissions set on the shares and files accessed through Dfs. | Minimizes administration workload, because the Windows 2000 file and directory permissions are used. |

**Table 6-1** Windows 2000 Dfs: Features and Benefits (continued)

| Feature | Description | Benefit |
|---|---|---|
| Intelligent client caching | When a user requests access to a shared folder through Dfs, the referral information for that portion of the namespace is saved in memory (cached) on the client. If the client requests access to the same portion of the namespace while that information is in cache, the client does not need to connect to the root server for the referral information. | Client caching of referral information reduces network traffic. |
| Dfs awareness for clients running Windows 9x | Dfs support is integrated into the redirector; therefore, no additional memory is required to run the Dfs clients for Windows NT 4.0 and Windows 98. An add-on component is needed to allow Windows 95 clients to access Dfs. | Because Dfs support is built into the Windows 98 and Windows NT redirector, no additional components need to be added. Windows 95 clients can have Dfs support added through the additional Dfs component that can be downloaded from the Microsoft site. |
| Interoperability with other network operating systems | If a shared folder is accessible through a redirector in Windows 2000, it can be part of the Dfs namespace, even if it is accessed through a server-based gateway. | Provides support for non-Windows 2000 operating systems, including MS-DOS, Microsoft Windows 3.1, Windows 9*x*, and Windows NT. |



## Dfs Terminology

Knowing the benefits of the product is great; however, you still need to know the terminology and mechanics of Dfs—forward and backward, we might add—if you are going to be an effective designer of Dfs structures to be used within your network.

We start our terminology exercise by introducing **Dfs topology**, which is the logical hierarchy of a Dfs, including the roots, links, shared folders, and replica sets, as depicted in the Dfs administrative console. Figure 6-2 shows the mapping of the Dfs topology, showing the links, and the servers and shares the links reference. Dfs maintains this topology information to redirect client requests to the appropriate place, but the topology is hidden from the user. What the user sees is the **Dfs namespace**, the logical view of shared folders on one or more servers, as presented by the Dfs client. The Dfs client interprets the topology information into this logical and simplified view.

A **Dfs root**, which is the logical starting point of a Dfs hierarchy and the top level of the namespace, is hosted on a server on what started out as an ordinary share. In Figure 6-2, the Dfs root is located on the server Corp1, and the root name is \\Corp1\SalesandMarketing. The last part of the name points to a share on the server, but the share could have a different

name than SalesandMarketing; in turn, the file folder that the share points to could have a different name entirely. The two links in Figure 6-2 are Point of Sales and Promotions, which point to shares on the servers Carmel and Liverpool, respectively.

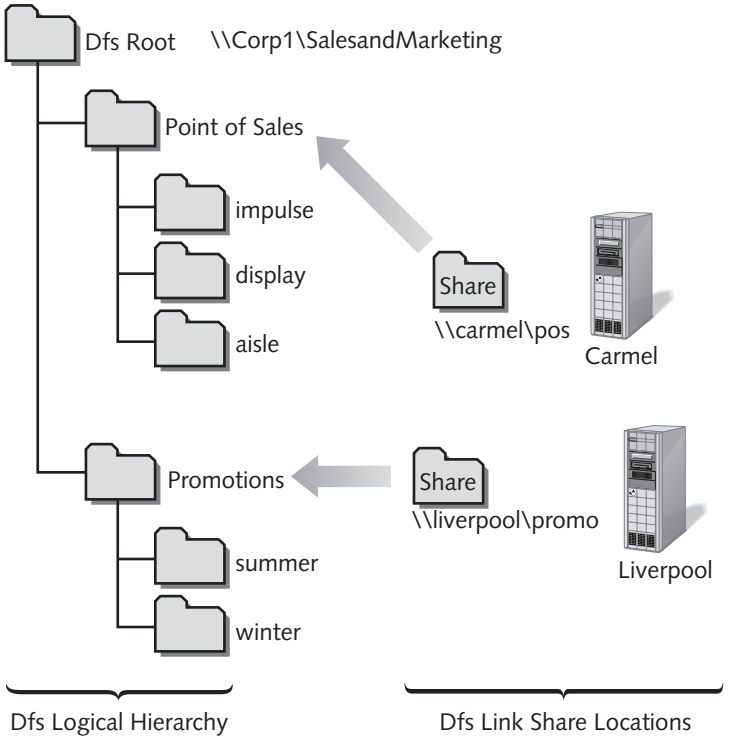


**Figure 6-2** Dfs topology mapping

Another basic Dfs term is **referral**, which, in Windows 2000, is information presented to a Dfs client attempting to gain access to a portion of the Dfs namespace. The referral contains a mapping of a DNS name to the UNC of the share associated with that portion of the Dfs topology.

Review time: DNS stands for Domain Name Service; UNC stands for Universal Naming Convention, which is a standard used in Microsoft networks for referencing a share on a server without using a drive mapping. Server Message Block (SMB) is Microsoft's file- and print-sharing protocol. The newest incarnation of SMB is Common Internet File System (CIFS).

With that brief introduction to the basic Dfs terminology, we now launch into an extended examination of the nuts and bolts of Dfs. We will look at the types of Dfs roots, the placement of Dfs roots, the addition of links, working with shared folders, working with replica sets, and understanding the Partition Knowledge Table (PKT).

## Types of Dfs Roots

There are two types of Dfs roots: standalone and domain. Standalone is Dfs as it was implemented for Windows NT 4.0, while domain Dfs is new in Windows 2000.

The **standalone Dfs root** is provided for downward compatibility with older Dfs clients, but it does not have the fault-tolerant capabilities of domain Dfs and cannot be located via a search of Active Directory. See Figure 6-3. Windows 2000 supports the standalone Dfs root that was supported by NT 4.0. In this type of Dfs, the Dfs root is hosted on a single computer and the Dfs topology is stored in the registry of the computer hosting the Dfs root. Unfortunately, this implementation provides no fault tolerance should the Dfs root server fail. Another limit of standalone Dfs is that there can only be a single level of Dfs links. This limitation is inconvenient because administrators may need to create a hierarchy of links.




You will create a standalone Dfs root in Hands-on Project 6-1.

Down-level Dfs clients can access a Windows 2000 standalone Dfs root. In this context, a **down-level Dfs client** is one running Windows 95, Windows 98, or Windows NT with the Dfs client software. Like Windows 2000, both Windows 98 and Windows NT (beginning with SP3) have the down-level Dfs client built-in. Windows 95 must have the Dfs client added to it. If any of these pre-Windows 2000 operating systems has the Directory Services Client installed, it can also access a domain Dfs root. Figure 6-3 shows what an administrator sees when viewing a standalone root server and links using the Distributed File System console. Notice that the Dfs root is represented in the tree pane as an ordinary UNC name (*servername**sharename*).

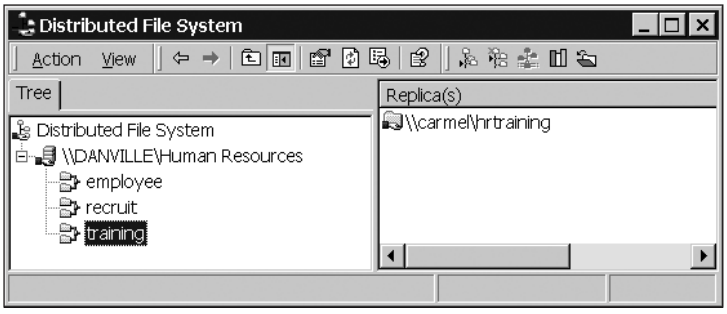


**Figure 6-3** Standalone root server and links

The second type of Dfs root is the **domain Dfs root**. In a domain Dfs root, the topology is stored in Active Directory and there can be multiple Dfs root servers—all required to be either domain controllers or member servers in the domain. If a root server fails,

you can restore a Dfs tree topology, because the Dfs topology is stored in Active Directory. Only Windows 2000 computers and computers running the Microsoft Directory Services Client, DSCLIENT.EXE, can access domain Dfs.

A domain Dfs minimizes unnecessary traffic because when a client requests access to a Dfs link, the file server that is closest to the client requesting the share will provide the share to the client. In addition, a domain Dfs root can have root-level Dfs shared folders, and it can support multiple levels of Dfs links.

The **Partition Knowledge Table (PKT)** for a domain Dfs is stored in Active Directory, and thus is available to each domain controller in a domain. The PKT maps root and replica nodes in the Dfs namespace to Active Directory sites and physical servers. Figure 6-4 shows the Dfs console view of a domain Dfs root and links. Notice the Dfs root represented in the tree pane by the domain name—intersales.corp—and the share name the administrator gave the root, SalesandMarketing.


You'll create a domain Dfs in Hands-on Project 6-5.

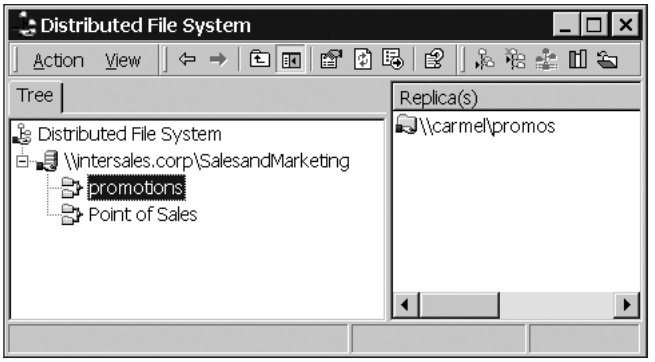


**Figure 6-4** Domain Dfs


Another name for a domain Dfs root is a fault-tolerant Dfs root. The Dfs console used this description in the beta versions of Windows 2000, but the final version refers to the root as a domain Dfs root. In this book, we will use this newer term, although you will still see references to fault tolerant Dfs in Microsoft documentation and other references.

## Placement of Roots

Now that you know the types of roots that are available, you need to concentrate on their placement. Placement of roots is an inexact science and a thankless job; if you do it right, no one notices. If you do it wrong, you get a great deal of negative attention

because people cannot access the data they require to accomplish their work. Poor root placement is sometimes known as a career-limiting opportunity.

If you are working with standalone roots, you have no choice but to attempt to locate the single standalone root as close to the users as possible. In the world of networking, "close" means on the same LAN (not across a WAN). Therefore, standalone Dfs roots should only be considered as a departmental or workgroup solution when there is no Windows 2000 domain.

If you have a Windows 2000 Active Directory domain, and you have users distributed across many sites who need access to Dfs volumes, consider placing a root server replica in each site. (You'll learn more about root replicas later in this chapter under "Replication of Dfs Roots.")



## Addition of Links

A **Dfs link** is a component of the Dfs topology located below the Dfs root. It forms a connection to one or more shared folders or to another Dfs root. To do this, it maps a DNS name to the standard UNC name of the shared folder to which it points. A Dfs root may have many links, each pointing to a different shared folder.


You'll create Dfs links in Hands-on Projects 6-2 and 6-6.

## Working with Shared Folders

A **Dfs shared folder** is a folder in the Dfs namespace that is shared by users with proper permission. Sharing of Dfs root-level folders is only supported in domain Dfs, but a share can be referred to by Dfs links in both types of Dfs.

## Working with Replica Sets

A **root replica** is a duplicate of a Dfs root on another server. It provides greater availability, because a root server is responsible for providing referrals to clients for shared folders. If a root server becomes unavailable and a root replica has not been created, the Dfs namespace becomes unusable.


You'll create root replicas in a domain-based Dfs in Hands-on Project 6-7.

### Understanding the Partition Knowledge Table (PKT)

The topology of a Dfs hierarchy is stored in the PKT. The PKT contains the Dfs directory name and the list of referral servers to which Dfs clients connect. On a standalone Dfs root, the PKT is stored in the root server's registry. In a domain Dfs root, the PKT is stored in Active Directory.

When a Dfs client selects a link from a Dfs root, it caches the portion of the Dfs namespace from that point down. When the client accesses a share below this link, it is actually "walking" down through its locally cached portion of the PKT. It caches this link information until the cache-referral period expires. This value is stored in the **Time to Live (TTL)** attribute for the root share or link share in the PKT. This saves network traffic. When the cache-referral period expires, if the client needs the referral, it reconnects to the Dfs root (or child) replicas to renew the cached information.

For those who like to do scripting of their administrative tasks, there are two command line utilities available for working with Dfs—DFSCMD.EXE and DFSUTIL.EXE. Although both can be used to manage standalone Dfs, only DFSUTIL.EXE is Active Directory-aware.

DFSCMD.EXE comes with Windows 2000 and can be found in the System32 folder under the system root. It is the same command that came with Dfs for NT 4.0. You can use DFSCMD.EXE to do essentially anything you can do with the Dfs administrative tool, except enable or disable file replication. DFSUTIL.EXE comes with the support tools on the Windows 2000 CD. It is an advanced tool for troubleshooting Dfs.


To gain further insight into these two command-line utilities, try Case Project #6-3.

## Dfs Processes and Network Activities

Dfs has several processes that can result in network activity. This activity is important to you because as a network designer, you must know when a network service you hope to use in your design generates network activity, so that you can measure that activity and estimate the load that it will place on the network. Only then can you plan for this as you evaluate the impact of your design decisions on the network traffic.

Of the Dfs processes, several are server-side, including those that maintain the PKT, replicate shared folders, switch between replicas during failover, and establish security. Other network activity is generated during interaction of the client with the server or servers involved in Dfs. These activities include establishing security, providing referrals to Dfs clients, gaining access to a Dfs shared folder, and linking logical names to physical addresses. We discuss all in turn next.

## Maintaining the PKT

The Dfs root server maintains a PKT containing a list of Dfs paths and share names. The server-based PKT stores the Dfs topology. All this stored information is valuable because work doesn't get done if the Dfs doesn't know where things are.

Table 6-2 shows the information stored in the PKT. In this table, \\danville\human resources is the Dfs root mapped to a share on Danville called hr. The second and third rows of the table list the links, and you can see that Dfs hides the actual path to each link. In this case, the actual shares reside on a server named Carmel.

**Table 6-2** Partition Knowledge Table

| Dfs Path | UNC Name | Caching Period for Referral |
|---|---|---|
| \\danville\hr | \\danville\humanresources | 300 seconds |
| \\danville\recruit | \\carmel\recruitment | 1800 seconds |
| \\danville\employee | \\carmel\employee | 1800 seconds |



## Caching Referrals by Clients

When a Dfs client accesses a Dfs root or link, the client computer caches a portion of the PKT. This portion is the referral list for the root or link. The Dfs client then connects to a server in the referral list. The client holds this referral list in memory for a predefined length of time, or "time to live." Then, if the client needs to connect again to the root or link within that time period, it does not have to generate any network traffic to pick up the referral list. It can simply use the location information in the cached referral to connect to a root or linked share. There is a separate caching setting for Dfs root referrals and for Dfs link referrals.

An administrator can modify the caching setting in the properties of each Dfs root and each Dfs link using the Dfs console. The default cache time on a Dfs root is 300 seconds, and the default cache time on a Dfs link is 1800 seconds. Figure 6-5 shows the properties page for a Dfs root, with the default client referral cache setting of 300 seconds.

Adjustments should be made to the caching period based on usage. For links in which the data changes frequently, shorten the caching period. For links in which the data changes infrequently, increase the caching period.

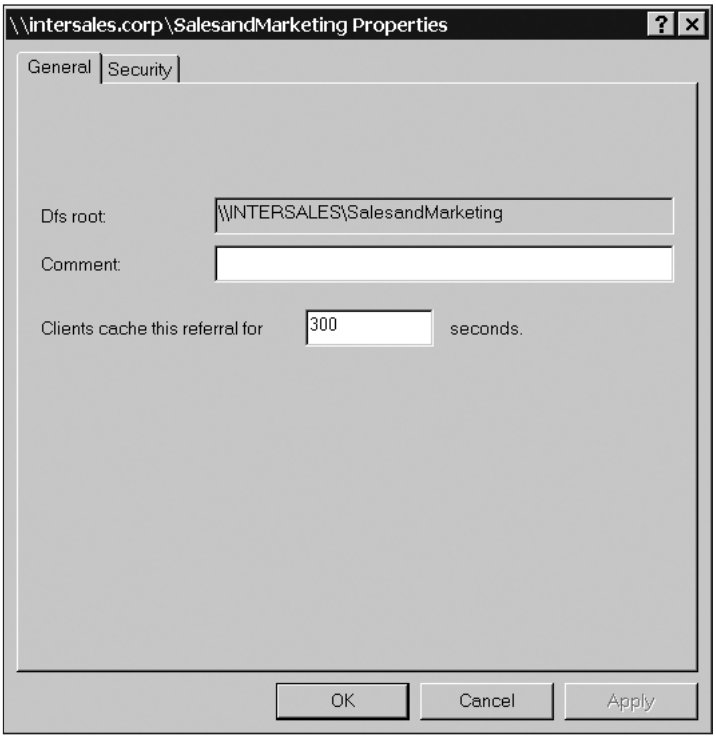


**Figure 6-5** Cache-referral setting for a Dfs root

## Gaining Access to a Dfs Shared Folder

When connecting to a domain Dfs root with root and **link replicas** (one of two or more shares pointing to the same link), a Windows 2000 client or a down-level client with the Directory Services client (DSCLIENT.EXE) installed receives a referral to a root server and link within its site. A non-directory-service–aware client will receive the list, but will randomize it and select a server from it without regard to Active Directory sites. This can cause traffic across sites.

In your design, you should consider that if you must have down-level clients that are not Active Directory aware, you should have them connect only to standalone Dfs roots. Your other option is to upgrade them to Windows 2000 or install the Directory Service Client (DSCLIENT.EXE) on them.

An administrator should frequently perform a status check on Dfs links to ensure that the links are still valid. This can be done through the Dfs console by right-clicking a root and selecting Check Status. The results are graphically a little underwhelming—just a circle with a green check on each share that is found to be up and running, as shown in Figure 6-6. But as we like to say, a little green check is better than a big blue screen.

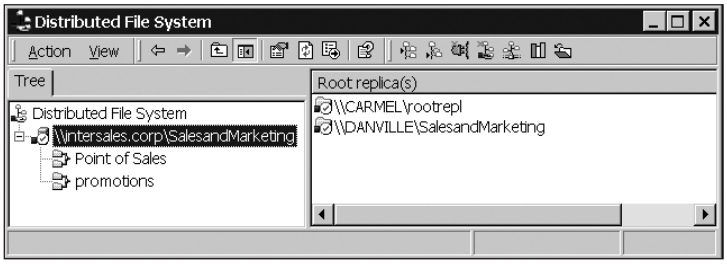


**Figure 6-6** Results of a status check



## Linking Logical Names to Physical Addresses

Users do not have to change their behavior to work with Dfs. Whether accessing shares normally or through Dfs, they can use the methods they have used in the past. This freedom means that a user with a Dfs client can access a Dfs volume using a UNC name or a drive mapping. In the case of the paths shown previously in Table 6-2, the user could have a drive mapped to the Dfs root and access the Dfs links as paths below the drive mapping, as shown in Table 6-3.

**Table 6-3** A Drive Mapped to a Dfs Root

| Dfs Path | UNC Path | Drive Mapping |
|---|---|---|
| \\danville\human resources | \\danville\hr | H:\ |
| \\danville\employee | \\carmel\employee | H:\employee |
| \\danville\recruit | \\carmel\recruitment | H:\recruit |
| \\danville\training | \\carmel\hrtraining | H:\training |

## Replication of Dfs Roots

A domain Dfs root can be replicated to one or more other servers. A standalone Dfs root cannot be replicated. Root replicas increase availability and fault tolerance. They increase availability because when an Active Directory-aware Dfs client attempts to contact the domain root, it will select the root that is in the same site as the client computer originating the request. It increases fault tolerance because if a server that is hosting a Dfs root replica fails, the client will be referred to another replica of the root. There is a limit of 32 replicas per domain root.

To create a new root replica, you would right-click the Dfs root and select New Root Replica. Figure 6-7 shows the Dfs console with the domain root \\intersales.corp\SalesandMarketing. Two replicas are shown in the Root replica(s) pane.

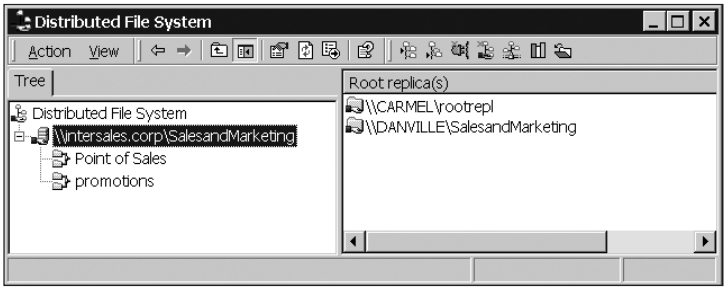


**Figure 6-7** Root replicas

## Replicating Shared Folders

To increase availability of data stored in Dfs, you can create replicas of links (up to 256 per root) for either a standalone or domain root. A replica is one of two or more shares pointing to the same link. A major shortcoming of link replicas in standalone Dfs is that the data must be replicated manually. This means that the administrator must figure out how to keep the data up to date on all the replicas manually. The point is that automated replication is not integrated with standalone Dfs.

On the other hand, if you create replicas of a Dfs link in a domain Dfs, you may choose to perform the replication manually or configure it to occur automatically. For the network design process, both types of replication have several procedures in common that you must document for the administrators who will carry out the ongoing administration:

- Determine where the source content comes from and how the administrators will acquire it.
- Determine how often the source content is updated.
- Establish and document which replica will be used as the master by the administrator to perform the updates.
- Once the content is placed on the master, initiate the replication.

In the case of manual replication, the administrator will have to establish the procedures for the actual replication of files. This need to replicate files to many servers is a common administrative problem (not just a Dfs issue) that server administrators often face. There are tools for automating this, like the NT Directory Replication Service (a.k.a. LMREPL or Lan Man Replication) for any NT servers involved in Dfs.

In the past, many administrators eschewed the NT Directory Replication Service for scripted solutions that used utilities such as Robocopy (NT Resource Kit). There are also many vendors of sophisticated data replication software such as Data Replication Solution by IBM (*www-4.ibm.com/software/data/dbtools/datarepl.html*), Itware-replicate from Insession Technologies (*www.insession.com/solutions/solutiondetail_itwarereplicate.asp*), and PowerSync by Link Pro (*www.linkpro.com*).

When you create a replica set in a domain Dfs, you must define the replication policy. The replication policy determines which computers and shares will participate in replication of the folder contents. In fact, the Replication Policy dialog box appears (after a short delay) after you add a replica to a link. As you can see in Figure 6-8, the default replication policy is "No," which means you are going to manually perform replication using scripts or specialized third-party software, as discussed previously.

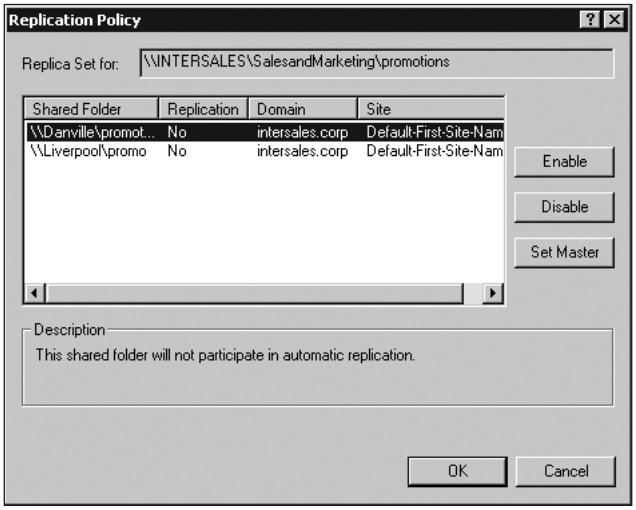


**Figure 6-8** Replication Policy dialog box with replication set to "No"

If you click the Enable button on the right side of the dialog box, the highlighted replica will participate in replication. Replication is provided by the Windows 2000 **File Replication Service (FRS)**, and it requires that each replicated share exist on an NTFS volume; therefore, it will not allow you to set replication to "Yes" if the share is not on an NTFS volume. The first replica for which you enable replication is by default the master, designated as "Primary" in the Replication Policy dialog box (Figure 6-9).

When you enable replication for another replica, as shown in Figure 6-10, the contents of the share on the primary replica will be replicated to the new replica. You can also define the primary replica using the Set Master button. The primary replica is only the master for the first replication. From this point on, all replicas are masters once they have replication turned on, so that a change to any one of the members is replicated to all other members.

For ongoing administration of **Dfs replication policy**, you can use the Replication Policy dialog box, available from the Distributed Files System console. Once you turn on automatic replication for each server in a replica set, you have done all the configuration of file replication that you can do from the Dfs console.

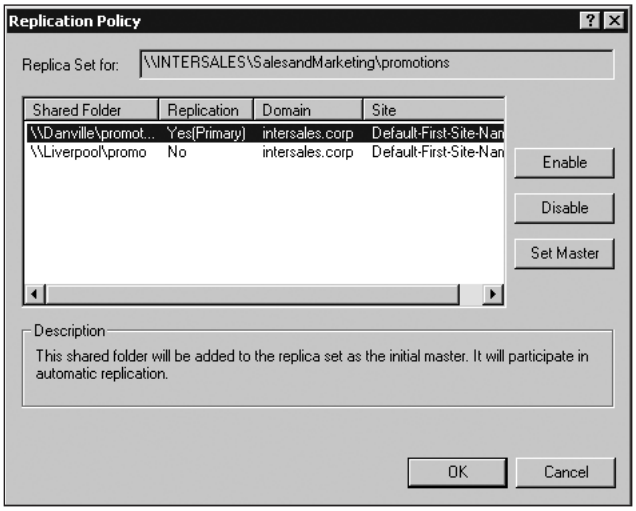


**Figure 6-9** Replication Policy dialog box with replication set to "Yes"

The FRS of Windows 2000 is a multi-master replication service that replaces the Directory Replication Service (a.k.a. LMREPL or Lan Man Replication) of NT 3.*x* and 4.0. Windows 2000 domain controllers and servers use FRS to replicate system policy and login scripts for Windows 2000 and down-level clients. As such, FRS is the underlying service that is used to replicate the content of Dfs roots or link replicas when you turn on automatic replication. FRS uses the "last writer wins" algorithm. This is a good reason to limit Dfs to read-only or infrequently changed data.

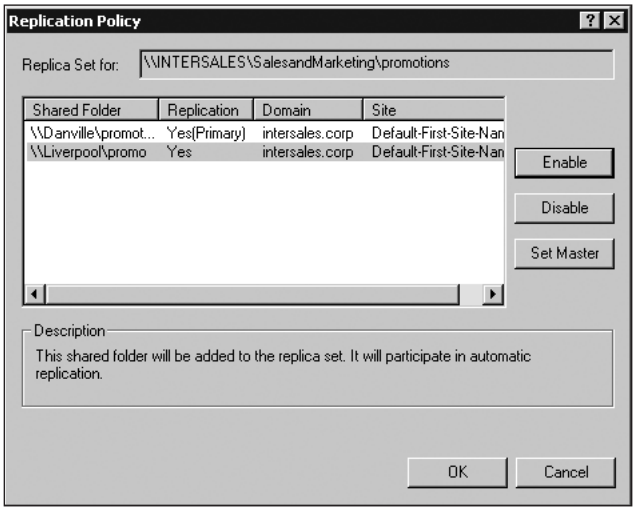


**Figure 6-10** Replication Policy dialog box with replication turned on for two replicas


To learn more about FRS and the "last writer wins" algorithm, see Microsoft Knowledge Base article Q221089.

FRS is somewhat more configurable for Dfs than it is for Active Directory information. We use the word "somewhat" because, unlike FRS replications for the Active SYSVOL, you can control what FRS replicates and when it replicates, you can specify a filter that excludes certain types of files or folders, and you can control scheduling of Dfs replication both within a site and between sites. Filters control exclusion of files and folders only for those files and folders added to a replica after the filter has been set. It is not retroactive; it will not remove existing files that match the filter.



Filters are set through the Active Directory Users and Computers console as follows:

1. Open the Active Directory Users and Computers console.
2. Click the View menu in the menu bar, and then click Advanced Features.
3. In the console tree pane, click System to expand it, expand File Replication Service, expand Dfs Volumes, and continue to expand folders until you reach the Dfs root and link to which you want to apply a filter.
4. Right-click the link, and click properties (see Figure 6-11, within which a link is selected and the properties sheet is displayed).
5. The first tab, Replica Set, has a file filter box in which you may add filters.
6. The Change Schedule button will allow you to change the replication schedule for this replica set.

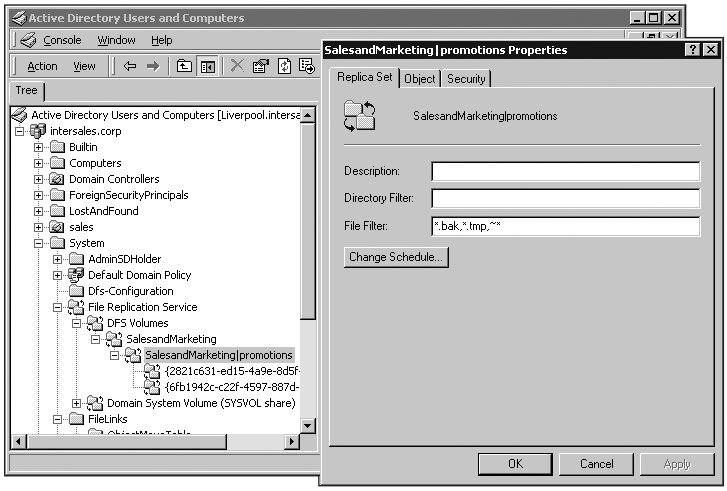


**Figure 6-11** Configuring FRS settings for Dfs replicas


If you plan to have root replicas, consider getting the post-Service Pack 1 patch to FRS. This patch addresses a problem with the service described in Microsoft Knowledge Base article Q265365. At the time of this writing, Service Pack 2 had not been released. This patch should also be part of Service Pack 2.

## Switching Between Replicas During Failover

If root replicas are used when a server hosting a root replica fails, Dfs directs clients to a replica root server. If links are replicated and a link becomes unavailable, Dfs will automatically direct requests to another replica.

## Establishing Security

Dfs does not enforce access control lists on the Dfs namespace, the Dfs roots, or the Dfs links. Thus, it depends on the permissions—both share permissions and NTFS permissions—set at the shared destination folder. This is a shortcoming, because share and file permissions have to be set in a separate operation (using a GUI tool or script) from Dfs creation and administration. Administrators cannot use a single tool (the Dfs console) to do it all.

If you want to simplify the administration of security for Dfs and you are using link replicas, automatic replication will help, but each share must reside on an NTFS volume. Set the permissions on the original shares before creating the links and before creating replicas. Then when you add replicas, and configure them to replicate automatically, be sure to designate the original share as the master (this is the default, but failure is only a mouse click away). Then the FRS will replicate the folders and files, and all shares and NTFS permissions will replicate.

To administer a standalone Dfs namespace, you need only be an administrator of the server hosting the Dfs root. To administer a domain Dfs namespace, you must be a member of the Domain Admins global group. To add a shared folder to a Dfs link, you do not need explicit permissions to the shared folder. However, users who access that folder must have permissions at the share and file system level that will permit access.

Fortunately, this lack of granularity in administering Dfs does not mean that delegation is unavailable for Dfs. You can centrally manage Dfs while delegating the administration of the underlying shares to people who are closer to the servers hosting the shares.

## How Dfs Finds Resources

Shares that you create for use by Dfs are still accessible outside of Dfs as ordinary shares. However, you do not want users to access them directly once you have Dfs configured. If you do allow users to bypass Dfs, you lose any advantages to be gained in availability and fault tolerance.

An understanding of how Dfs finds resources will underline the importance of using the Dfs names when connecting to the resources. Consider this scenario: You use Windows Explorer to map a drive to connect to the Dfs volume \\intersales.corp\salesandmarketing\promotions. This is a Dfs share accessed through a link to the domain Dfs root hosted on a member server named Danville.

In this scenario, your redirector will be used first in an attempt to locate this resource. If you only have the Microsoft redirector, it won't take too long to figure out that intersales.corp is not the name of a server. But if you have multiple redirectors, such as the NetWare redirector of the Gateway or Client services for NetWare, it could take longer, because each redirector will be given the opportunity to find the resource.



Once the redirectors fail, and they will fail, the Multiple UNC Provider (MUP.SYS) has a chance at finding the resource. Dfs is actually a part of the MUP, and as such, it contacts Active Directory, and the Dfs service provides the location of the Dfs root from which the Dfs client gets a referral to the share for Promotions. In this case, it actually points to a share that is yet another domain root with links below it.

The moral of the story is that you really need to know the technology before you use it, both how it works and how it breaks. The knowledge you are gaining will protect you from making serious mistakes in your designs.

## Dfs Design Strategies

Well, if you've made it this far, you are pretty adept at moving through the intricacies of this product. Now comes the fun part—design strategies. There are, as always, several areas to pay attention to when designing elements of a Windows 2000 network infrastructure. These are functionality, security, availability, and performance. We will discuss design strategies for each of these in turn.

### Functional Dfs Design

Dfs should be considered in your Windows 2000 network infrastructure design if any of the following conditions exist:

- Users who need access to the same file resources are distributed across multiple sites.
- A majority of users need access to multiple shared file resources.
- User access to shared file resources must be highly available (24/7).
- Data is stored on many distributed network shares.
- Data files are read-only or rarely modified by just a few individuals.

If one or more of these conditions exist, then Dfs is a solution for your network. Let's expand on this. Consider a large national electrical contractor. Their estimators, designers, and engineers must have access at all hours to stores of electrical codes, specifications, and regulations. At each of their 200 sites are servers hosting shares containing these data stores. The information is updated from a central server once a month. They need the information at each site so that they have it available even if a WAN link goes down. They are presently using standard SMB (LAN Manager) shares on their NT servers, and administrators are using an awkward combination of Robocopy and a script they run each month to automate this process.

They have upgraded their domain to Windows 2000 Active Directory and purchased the equipment to upgrade each of the file servers hosting the data stores. This scenario fits most of the criteria above, and it is not an unusual situation. Once they replace their NT servers with Windows 2000 servers and have them join the domain, they can create a domain Dfs root at their central location and create the first link. A strategy we like is to place the share referenced by this link on the root server itself, and then copy the data onto this share. Do this for each share you wish to create that will contain unique data.

Once you have created one link for each set of information, then create replicas for each of those on one of the Windows 2000 member servers in the remote offices (you can do this centrally from the Dfs console). Select Automatic Replications, and be sure that the central one (where you have placed all the information) is the master. They will replicate. The master is only needed for the first replication. With all subsequent replicas that you add to each replica set, you will not have to designate a master. Continue to place a replica on a server at each site that must have access to that data. Now when you update the information monthly, you only need to copy it to any of the servers in a replica set. It will replicate to all the other replicas.

Dfs is also a solution for people who must publish data through a Web server. Here again, this is best if it is read-only data, even with replicas and the automatic replication of FRS. The IIS server, which is the server that responds to client requests for Web and FTP pages, does not have to actually host the files that it is serving up. It can be configured to point to a UNC name that could be a Dfs root. With a domain root and replicas at the root and link level, you would have load balancing and fault tolerance for the IIS data.

The downside to the Dfs/IIS strategy is that it is actually the IIS server that accesses the data over on the share with an account and password the IIS administrator has configured for connection to the share. Therefore, you cannot have any granularity of access to that data. In other words, this works best for anonymous access, not for personalized user authenticated access. Although it is *way* beyond the scope of this book, we recommend that you check out Microsoft Application Server if you need the type of functionality we are trying to achieve using Dfs with IIS.

You also should consider using Dfs for source files for server-based application installations. This works very nicely with group policy–initiated Windows installer packages. When setting this up, place the source files on a share to which you create a Dfs link. In addition, for availability and fault tolerance, create replicas of the links.

Now that you have considered the conditions and scenarios that might influence you to create a Dfs solution for your network design, you also need to develop naming strategies for the root and links involved in your design. We will explore this topic next.

### Naming Strategies



Remember that to the user, the Dfs root and links will appear as ordinary shares that they may continue to access in the manner to which they are accustomed. Thus, as a best practice, you should create the shares that will become the root shares by using short descriptive names. These will normally only be used by administrators, but the names users will see as part of their Dfs namespace—the Dfs root and link names—should be short and descriptive of the purpose or content of the shares.

In our own test and design lab, we have used a Dfs root folder named SalesandMarketing; although it is not the shortest name, it is descriptive of the department that will be accessing this Dfs hierarchy. This is useful simply because users recognize their department name. The links below it are named Point of Sales and Promotions, which have special meaning to the users who will be accessing these resources. Folders below Promotions include Summer and Winter. Figure 6-12 shows a functional Dfs design.

## Secure Dfs Design

To have a secure Dfs design, you must set the appropriate permissions on the shares and NTFS folders and files. In the case of domain Dfs replicas on NTFS volumes, set these permission on one replica, and they will automatically be replicated to the other replicas.

Be careful to plan for how and when modifications should be made to replicas. The rule with FRS is the last write wins; thus, even with FRS replications, Dfs should only be considered a solution for read-only data. In addition, you should adjust the permissions so that only administrators can make modifications, and then make sure that the administrators coordinate changes to the replicas to avoid confusion when one administrator overwrites the changes of another administrator.

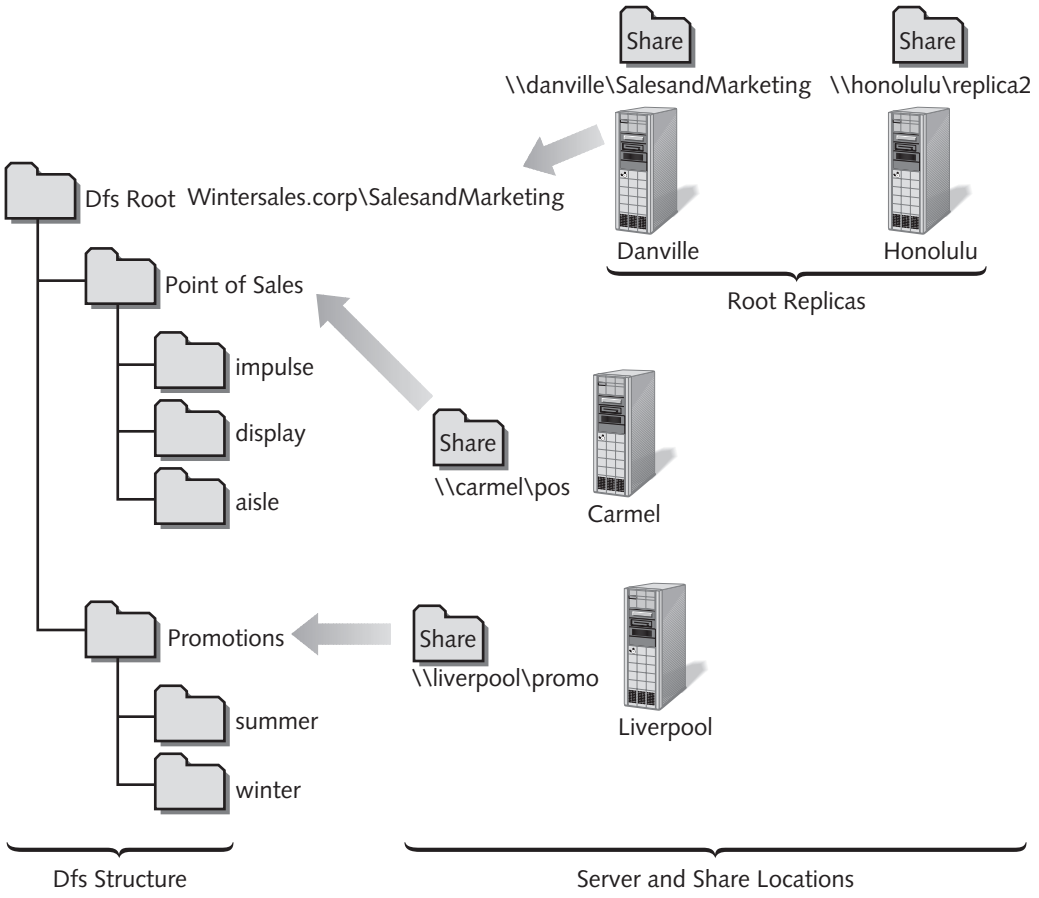


**Figure 6-12** A functional Dfs design

## Enhancing the Dfs Design for Availability

Strategies to enhance a Dfs design for availability depend on whether you are working with a standalone or domain Dfs. Let's explore this a little:

- A standalone root can be enhanced by the creation of link replicas. Standalone Dfs does not support automatic replication, but if the data is truly read-only, this should not be a huge problem.
- However, with standalone Dfs, the root is a point of failure. A solution to this is to use a Windows 2000 Server Cluster, in which you define the Cluster File Share to be a standalone Dfs root. Dfs clients will then access the Dfs links with one mapping to a virtual server, which will be serviced by either of the servers in the cluster. If one server fails, the other will respond.

- The best strategy to improve availability of files that users in many sites need (if you have an Active Directory domain) is Domain Dfs and all its features. Place both root replicas and link replicas in each site and enable automatic replication to make resources available even when WAN links go down.


See Microsoft Knowledge Base article Q220819 for more details on Windows 2000 Server Cluster. See the "Microsoft Windows 2000 Server Distributed Systems Guide" of the Windows 2000 Server Resource Kit for more information on Dfs.

## Enhancing the Dfs Design for Performance



There are several actions that you can take to enhance a Dfs design for performance:

- First of all, look at the servers involved in the Dfs tree. Dfs bottlenecks are most likely to occur at the disk and network card; therefore, be sure that you have a fast high-end network card and a fast disk subsystem in the servers.
- The steps you take to enhance availability of Dfs will also boost performance in a Dfs design. For instance, good placement of servers can enhance a design for performance, especially when you are working with domain Dfs.
- In addition, whenever you are designing a Dfs solution that spans locations and you have a Windows 2000 domain, use domain roots and create and configure replicas at both the root and link level. This action places servers containing root replicas and link replicas in each site where users who need the data reside.
- Be sure to enhance the previous strategy with automatic replication.


If you want really fast performance on a LAN, you can have a bunch of Dfs servers that give load balancing client access. Configure multiple NICs in each server and connect one NIC from each server to a standalone switch, and then configure the replication and other server-to-server traffic to use that NIC as a dedicated private network. This way, replication and other traffic won't conflict with client requests. Make sure to put these NICs and the backup system on different PCI buses in your server to minimize internal contention. Also, consider RAID 0 rather than RAID 5, since the data is always replicated on several other servers.

Your end result will be a distributed file management design that provides the data that users need, where and when they need it, at a level of security they require, and at a level of performance they require and desire.

## Chapter Summary

We began this chapter by examining how network usage has moved from departmental LANs, with most network traffic confined locally to the distributed computing of client/server environments. Today, users are accessing data residing on many servers throughout the enterprise. This has led to the need for a Dfs that can make resources hosted on many servers available to users within a hierarchical namespace that appears logically coherent.

You reviewed a summary of the features and benefits of Windows 2000 Dfs followed by an examination of Dfs fundamentals. This included the two types of Dfs roots: standalone and domain. The Dfs topology, or logical hierarchy, was viewed in the Dfs console with all of its components—including the roots, links, shared folders, and replica sets.

The Dfs processes were defined next. This included maintaining the PKT, caching referrals, gaining access to Dfs resources, linking logical names to physical addresses, replicating Dfs roots and links, and switching replicas during failover. We also defined certain administrative tasks, including establishing security at the file system and share level, checking on the status of Dfs shares, creating replicas, configuring replication, and working with command-line utilities.

Finally, we defined the qualities of a functional Dfs design, and then considered how to enhance the design for security, availability, and performance.

## Key Terms

**Dfs link** — Defined on a Dfs root and appearing to users as a folder below the Dfs root, it is a pointer to a share on that server or another server. It can also point to another Dfs root.

**Dfs namespace** — The logical view of shared resources seen by users from Dfs client computers.

**Dfs replication** — Replication of the root files and folders between root replicas or between Dfs link replicas. Replication is provided by the Windows 2000 File Replication Service (FRS), which is only supported in domain-based Dfs.

**Dfs root** — The logical starting point of a Dfs hierarchy, hosted on a server.

**Dfs shared folder** — A folder in the Dfs namespace that is shared by users with proper permission. Sharing of Dfs root-level folders is only supported in domain-based Dfs, but a share can be referred to by Dfs links in both types of Dfs.

**Dfs topology** — The logical hierarchy of a Dfs, including the roots, links, shared folders, and replica sets, as depicted in the Dfs administrative console.

**Distributed File System (Dfs)** — A distributed file management system that creates a unified namespace for users, although the folders of the namespace may reside on many different servers on the network. Users of Dfs simply see a share on a server with a folder hierarchy beneath it. It does not look any different to the user than an ordinary share with disk folders beneath it.

**Domain Dfs root** — In a domain Dfs root, the topology is stored in Active Directory and there can be multiple Dfs root servers, all required to be either domain controllers or member servers in the domain. This is in contrast to a standalone Dfs root in which the topology is stored in the registry of the Dfs root server.

**down-level Dfs client** — A computer running an operating system previous to the current operating system and running the Dfs client appropriate to that operating system.

**File Replication Service (FRS)** — A replication service available on Windows 2000 servers in an Active Directory domain. This service replaces the LMRepl service of Windows NT. FRS only works on NTFS volumes, and can only be used in conjunction with other services, such as Active Directory and Dfs.

**link replica** — One of two or more shares pointing to the same link.

**Partition Knowledge Table (PKT)** — A table that maps links and shares for the Dfs namespace.

**root replica** — A duplicate of a Dfs root on another server, it provides greater availability because a root server is responsible for providing referrals to clients for shared folders. If a root server becomes unavailable and a root replica has not been created, the Dfs namespace becomes inoperative.

**referral** — In Dfs for Windows 2000, it is information presented to a Dfs client attempting to gain access to a portion of the Dfs namespace. The referral contains a mapping of a DNS name to the UNC of the share associated with that portion of the Dfs topology.

**standalone Dfs root** — Windows 2000 supports the standalone Dfs root supported by Windows NT 4.0, in which the Dfs root is hosted on a single computer and the Dfs topology is stored on that computer.

**Time to Live (TTL)** — When a Dfs client gains access to a shared folder in the Dfs namespace, it caches that portion of the table for the length of time specified in the TTL attribute for the root share or link share.



## Review Questions

1. You manage over 200 servers that host shares containing read-only data that you update monthly. At present, each server has ordinary file and print (SMB) shares, and the logon scripts for the users contain a drive mapping to the shares. Whenever you move the data to a new server, it causes calls to the help desk, because users cannot find their data in the same old place. How would the use of Dfs improve this situation?
2. Dfs is a new technology in Windows 2000. True or False?

3. A Windows 2000 server can host which of the following types of Dfs roots? (Select all that apply.)
   a. secure Dfs root
   b. standalone Dfs root
   c. independent Dfs root
   d. domain Dfs root
   e. integrated Dfs root
4. Describe five features of Windows 2000 Dfs and the benefits of each.
5. Define Dfs topology.
6. Define Dfs namespace.
7. The logical starting point of a Dfs hierarchy is a _____.
   a. domain controller
   b. member server
   c. Dfs link
   d. Dfs root
   e. Distributed Files System console
8. What two steps could you take to provide fault tolerance for a Dfs namespace with a standalone root?
9. Of the following, which can host a domain Dfs root? (Select all that apply.)
   a. Windows 2000 domain controller
   b. NT 4.0 BDC in a Windows 2000 domain in mixed mode
   c. Windows 2000 member server in a Windows 2000 domain
   d. Windows 2000 Professional member of a Windows 2000 domain
   e. Windows 2000 Server member of a workgroup
10. Where is the Dfs topology stored for a domain Dfs?
11. What clients can access domain Dfs?
12. Describe how a domain Dfs can minimize network traffic.
13. Define Dfs link.
14. Define a root replica and give its principal benefit.
15. What information is in the PKT?
16. What two command-line utilities can be used to create and administer Dfs?
17. How can an administrator modify the length of time a client caches the referral information for a Dfs link, and what is the default cache period?
18. How can an administrator verify that a link is still valid on the network?
19. What service is used to replicate the contents of replicas in a domain Dfs?

20. When you create a replica set in a domain, Dfs replication will occur automatically. True or False? Explain your answer.
21. What tool would you use to set a file filter for a replica set and modify the schedule for Dfs replication of a replica set?

## HANDS-ON PROJECTS


To do these hands-on projects, you will need a computer that is running Windows 2000 Server or Advanced Server and that is a member of the Intersales.corp domain. The computer should have both a C: and a D: partition.




### Project 6-1 Creating Shares for Dfs and Creating a Dfs Standalone Root

Even with the new capabilities of the domain-based Dfs in Windows 2000, there will be times when a standalone Dfs will be more appropriate. Perhaps you have no Windows 2000 domain, or your down-level clients are not running the Directory Service Client (DSCLIENT.EXE), or you want to limit the scope of a Dfs hierarchy to a certain area of your enterprise.

This project is broken up into two parts: 6-1A, in which you create shares on your server to use for Dfs; and 6-1B, in which you create a Dfs standalone root on your server. In Project 6-1B you will be using the Dfs console from the Administrative Tools folder. If this folder is not available from your start menu, or if Dfs is not available from the Administrative Tools menu, complete the Optional Hands-on Project.

To execute Hands-on Project 6-1A:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete**.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Computer Management**.
8. Double-click **Shared Folders** under System Tools in the Computer Management console.
9. Right-click **Shares**.

10. Click **New File Share**.
11. In the Create Shared Folder Wizard, click the **Browse** button.
12. In the Browse For Folder page, click **Local Disk (C:)**.
13. Click the **New Folder** button. A new folder appears in the Browse For Folder window with the name New Folder.
14. Type **dfsroot** for the new folder name and press **Enter**.
15. Click the **OK** button.
16. In the Create Shared Folder Wizard, verify that **C:\dfsroot** appears in the Folder to Share box. If the specified folder name does not appear, browse to it.
17. In the Share Name box, type **Human Resources**.
18. In the Share Description box, type **Root Dfs for Human Resources**.
19. Click the **Next** button.
20. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
21. Click the **Finish** button.
22. Read the Create Shared Folder message that appears.
23. Click the **Yes** button, because you need to create another share.
24. Click the **Browse** button next to the Folder to Share box.
25. Click **Local Disk (D:)**.
26. Click the **New Folder** button. A new folder appears.
27. Type **Recruit** to replace New Folder as the name for the folder and then press **Enter**.
28. Click the **OK** button.
29. Back in the Create Shared Folder Wizard, verify that **D:\recruit** appears in the Folder to Share box. If the specified folder name does not appear, browse to it.
30. Click the **Share Name** box, and then type **recruitment**.
31. Click the **Share Description** box.
32. Type **Recruitment procedures and Info**.
33. Click the **Next** button.
34. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
35. Click the **Finish** button. The Create Shared Folder message appears.
36. Click the **Yes** button, because you need to create another share.
37. Click the **Browse** button next to the Folder to share box.
38. Click drive **Local Disk (D:)**.
39. Click the **New Folder** button. A new folder appears.

40. Type **employee** to replace New Folder as the name for the folder, and then press **Enter**.
41. Click the **OK** button.
42. Back in the Create Shared Folder Wizard, verify that **D:\employee** appears in the Folder to Share box. If specified folder name does not appear, browse to it.
43. Click in the **Share Name** box, type **employee**, and then click the **Share Description** box.
44. Type **Employee records**.
45. Click the **Next** button.
46. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
47. Click the **Finish** button. The Create Shared Folder message appears.
48. Click the **No** button because you do not need to create another shared folder for this project.
49. The Computer Management console appears. Verify that your new shares are displayed under Shares.
50. Close the Computer Management console.



If you are already logged on to your server as administrator from the Intersales.corp domain, you may skip to Step 7.

To execute Hands-on Project 6-1B:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. In the Dfs Console, right-click **Distributed File System**.
9. Click **New Dfs Root**.
10. On the Welcome page of the New Dfs Root Wizard, click the **Next** button.
11. On the Select the Dfs Root type page of the New Dfs Root Wizard, select **Create a standalone Dfs root**.

12. Click the **Next** button.
13. In the Specify the Host Server for the Dfs Root page of the New Dfs Root Wizard, verify that the **Server name:** text box displays the name of your server: *yourserver*.intersales.corp.
14. Click the **Next** button.
15. In the Specify the Dfs Root Share page of the New Dfs Root Wizard, verify that **Use an existing share** is selected and use the down arrow to the right in the box to select the share to be used as the root. Select **Human Resources**, the share you created in the preceding steps.
16. Click the **Next** button. The Name the Dfs Root page of the New Dfs Root Wizard appears with the path to the share on your server and the Dfs root name.
17. In the Comment box, type **Root share for Human Resources**.
18. Click the **Next** button.
19. On the Completing the New Dfs Root Wizard page, verify that the **Host server** box displays the DNS name of your server and that the root share and Dfs root names are correct.
20. Click the **Finish** button. In the Dfs console, your server should appear with the root share. Notice that the server name is included. This is an indication that it is a standalone root. You now have a standalone Dfs root for the human resources department.
21. Close the Dfs console.

To make Dfs usable, you must create links to shares where data resides. You will add links to this Dfs root in Hands-on Project 6-2.


## Project 6-2 Creating Dfs Links to a Standalone Dfs Root

This project depends on the successful completion of Hands-on Project 6-1. After you have created a Dfs root, you will want to create Dfs links. Although you will usually be creating these links to other servers on your network, in this lab, you will create links from the DFSROOT you created on your drive C: to the shares you created on drive D:.

In the following steps, you will add a Dfs link to the Dfs root on your server. If you are already logged on to your server to the Intersales.corp domain, you may skip to Step 7.

To execute Hands-on Project 6-2A:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)

6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. If your Dfs root does not appear in the console, do Steps 9 through 12; if it does appear in the console, you may skip to Step 13.
9. Right-click **Distributed File System** in the Tree pane of the console window.
10. Click **Display an Existing Dfs Root**.
11. In the Dfs root or host server box, type: ***yourserver*\human resources**.
12. Press the **Enter** button. The your*servername*\human resources will appear next to a shared server icon in the left pane.
13. Right-click **your server**.
14. Click **New Dfs Link**.
15. In the Create a New Dfs Link dialog box, type **recruit** in the Link name box.
16. Click the **Browse** button.
17. In the Browse for Folder box, double-click **Entire Network**.
18. Double-click **Microsoft Windows Network**.
19. Double-click **Intersales**.
20. Double-click **yourserver**.
21. Double-click **recruitment**, and then click **OK**.
22. Click the **OK** button on the Create a New Dfs Link page. The Dfs console appears.
23. Verify that the new link appears under your Dfs root.

In the following steps, you will add another Dfs link to the Dfs root on your server.

To execute Hands-on Project 6-2B:

1. Right-click **your server**.
2. Click **New Dfs Link**.
3. In the Create a New Dfs Link dialog box, enter **employee** in the Link name box.
4. Click the **Browse** button.
5. In the Browse for Folder box, double-click **Entire Network**.
6. Double-click **Microsoft Windows Network**.
7. Double-click **Intersales**.
8. Double-click ***yourserver***.
9. Double-click **employee**.
10. Click the **OK** button.
11. Click the **OK** button on the Create a New Dfs Link page. The Dfs console appears.

12. Verify that the new link appears under your Dfs root.
13. Close the Dfs console.

You have created a Dfs link to shares on the same server hosting the Dfs root. Normally, you would create links pointing to shares on remote servers.


## Project 6-3 Removing a Standalone Dfs Root

You may need to remove a Dfs root when making changes to servers. If you completed Hands-on Project 6-1, you have a standalone Dfs root on your server. A server may only host a single Dfs root, regardless of the type of Dfs root. Since you will be creating a domain-based root on your server in Hands-on Project 6-4, you must remove the standalone Dfs root from your server.

Follow these steps to remove a Dfs root. (*Reminder*: This project depends on the completion of Hands-on Project 6-1.) If you are already logged on to your server as administrator from the Intersales.corp domain, you may skip to Step 7.

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security dialog box titled **Log On to Windows**.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. In the Dfs console, right-click the domain root you created in Hands-on Project 6-1.
9. Click **Delete Dfs Root**.
10. Click the **Yes** button.
11. Close the Dfs console.

You have successfully removed a Dfs root from your server. Now your server is available to host a new Dfs root.


## Project 6-4 Preparing for a Domain-Based Dfs

When designing a Dfs strategy, you will want to use a domain-based Dfs to take advantage of the fault-tolerance capabilities and also to provide availability. In this project, you will be using the Dfs console from the Administrative Tools folder. If this folder is not available from your start menu, or if Dfs is not available from the Administrative Tools menu, complete the Optional Hands-on Project for this chapter.

In this project, you create shares to use as the root share and links. Then in Hands-on Project 6-5, you will create a domain-based Dfs root on a Windows 2000 server and create links to that root server. If you are already logged on to your server as administrator from the Intersales domain, skip to Step 7.



To create shares to use for the Dfs root share and the Dfs links:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security dialog box titled Log on to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Computer Management**.
8. Double-click **Shared Folders** under System Tools in the Computer Management console.
9. Right-click **Shares**.
10. Click **New File Share**.
11. In the Create Shared Folder Wizard, click the **Browse** button.
12. In the Browse For Folder page, click **Local Disk (C:)**.
13. Click the **New Folder** button. A new folder appears in the Browse For Folder window with the name New Folder.
14. Type **domainroot** to replace New Folder as the name for the folder and then press **Enter**.
15. Click the **OK** button.
16. In the Create Shared Folder Wizard, verify that c:\domainroot appears in the Folder to Share box.
17. In the Share Name box, type: **SalesandMarketing**.
18. In the Share Description box, type: **Root dfs for SalesandMarketing**.

19. Click the **Next** button.
20. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
21. Click the **Finish** button. The Create Shared Folder message appears.
22. Click the **Yes** button, because you need to create another share.
23. Click the **Browse** button next to the Folder to share box.
24. Click **Local Disk (D:)**.
25. Click the **New Folder** button. A new folder appears.
26. Type **promotions** to replace New Folder as the name for the folder, and then press **Enter**.
27. Click the **OK** button.
28. Back in the Create Share Folder Wizard, verify that **D:\promotion** appears in the Folder to Share box. If the specified folder name does not appear, browse to it.
29. Click the **Share Name** box.
30. Type **promotions**.
31. Click in the **Share Description** box.
32. Type **Promotion campaign data**.
33. Click the **Next** button.
34. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
35. Click the **Finish** button. The Create Shared Folder message appears.
36. Click the **Yes** button, because you need to create another share.
37. Click the **Browse** button next to the Folder to Share box.
38. Click **Local Disk (D:)**.
39. Click the **New Folder** button. A new folder appears.
40. Type **pos** to replace New Folder as the name for the folder, and then press **Enter**.
41. Click the **OK** button.
42. Back in the Create Shared Folder Wizard, verify that **D:\pos** appears in the Folder to Share box. If the specified folder name does not appear, browse to it.
43. Click the **Share Name** box.
44. Type **Point of Sale**.
45. Click the **Share Description** box.
46. Type **Point of Sale data**.
47. Click the **Next** button.

48. In the permissions page of the Create Shared Folder Wizard, select **Administrators have full control; other users have read-only access**.
49. Click the **Finish** button. The Create Shared Folder message appears.
50. Click the **No** button because you do not need to create another shared folder.
51. The Computer Management console appears. Verify that your new shares are displayed under Shares.
52. Close the Computer Management console.


## Project 6-5 Creating a Domain-Based Root on Your Server



If you are already logged on to your server as administrator from the Intersales domain, skip to Step 7.

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. In the Dfs console, right-click **Distributed File System**.
9. Click **New Dfs Root**.
10. On the Welcome page of the New Dfs Root Wizard, click the **Next** button.
11. On the Select the Dfs Root Type page of the New Dfs Root Wizard, select **Create a domain Dfs root**.
12. Click the **Next** button. The Select the Host Domain for the Dfs Root page of the New Dfs Root Wizard appears.
13. Verify that Intersales.corp is in the Domain Name box, and then click the **Next** button.
14. Verify that *yourserver*.intersales.corp is in the Server Name box. If not, type it in the box.
15. Click the **Next** button.
16. On the Specify the Dfs Root Share box, verify that Use an existing share is selected, and then use the down arrow button to the right in the box to select **SalesandMarketing**, the share you created earlier.

17. Click the **Next** button. The Name the Dfs Root page of the New Dfs Root Wizard appears.
18. In the Dfs Root Name box, verify that **SalesandMarketing** appears.
19. Click the **Next** button. In the Completing the New Dfs Root Wizard page, verify that the following information appears, (use the back button to make any necessary corrections):
    Domain: **intersales.corp**
    Host server: ***yourserver*.intersales.corp**
    Root share: **SalesandMarketing**
    Dfs root name: **SalesandMarketing**
20. Click the **Finish** button.
21. Notice that in the Dfs console the domain root appears as **\\intersales.corp\salesandmarketing**.
22. Close the Dfs console.


## Project 6-6 Creating Dfs Links to a Domain-Based Root

This project depends on the successful completion of Hands-on Project 6-5. After you have created a Dfs root, you will want to create Dfs links. In the real world, you will usually be creating these links to other servers on your network. However, in this lab you will create links from the SalesandMarketing Dfs root share you created on your Local Drive C to the shares you created on your Local Drive D. If you are already logged on to your server as administrator in the Intersales.corp domain, skip to Step 7:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. If your Dfs root does not appear in the console, perform Steps 9 through 12. Otherwise, skip to Step 13.
9. Right-click **Distributed File System** in the Tree pane of the console window.
10. Click **Display an Existing Dfs Root**.

11. In the Dfs Root or Host Server box, type: ***intersales.corp*\SalesandMarketing**.
12. Press **Enter**. The domain root will appear as \\intersales.corp\SalesandMarketing next to a shared server icon in the left pane.
13. Right-click the **root** (\\intersales.corp\SalesandMarketing).
14. Click **New Dfs Link**.
15. In the Create a New Dfs Link dialog box, enter **promotions** in the Link name box.
16. Click the **Browse** button.
17. In the Browse For Folder box, double-click **Entire Network**.
18. Double-click **Microsoft Windows Network**.
19. Double-click **Intersales**.
20. Double-click ***yourserver***.
21. Double-click **promotions**.
22. Click the **OK** button. The Create a New Dfs Link page appears. Notice the client cache referral value of 1800 seconds. You may modify this here or in the Dfs console after the link is created.
23. Click the **OK** button on the Create a New Dfs Link page.

In the following steps you will add another Dfs link to the Dfs root on your server:

24. Right-click your server.
25. Click **New Dfs Link**.
26. In the Create a New Dfs Link dialog box, enter **Point of Sales** in the Link name box.
27. Click the **Browse** button.
28. In the Browse For Folder box, double-click **Entire Network**.
29. Double-click **Microsoft Windows Network**.
30. Double-click **Intersales**.
31. Double-click ***yourserver***.
32. Double-click **Point of Sale**.
33. Click the **OK** button.
34. Click the **OK** button in the Create a New Dfs Link dialog box.
35. Close the Dfs console.

You have now created two Dfs links to shared folders on a volume on the same server hosting the Dfs root. Again, you normally would create Dfs links to shares on other servers on the network.


## Project 6-7 Creating a Dfs Root Replica

In this project, you will create a Dfs root replica. For this project, you will need a lab partner. One lab partner, Partner A, must have completed Hands-On Projects 6-5 and 6-6 on his or her server, which we will call Server A, and the server must have a domain-based Dfs root. The other lab partner, Partner B, must not have a Dfs root on his or her server, which we will call Server B. If Server B has a Dfs root, it must be removed. Instructions for removing a Dfs root are in Hands-on Project 6-3. Once Partner B has removed the Dfs root, proceed with the lab. Partner B will do the following set of steps to create a share to be used as a root replica. If Partner B is already logged on to Server B, Partner B can skip to Step 7.

### Partner B only:

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, right-click **My Computer**.
8. Click **Explore**.
9. Double click **C:**.
10. Click the **File** menu in the menu bar of the window.
11. Point to **New**, and then click **Folder**. This will create a new folder in the root of drive C:.
12. Type **rootrepl** to replace New Folder as the name for the folder.
13. Press **Enter**. The folder rootrepl should now exist in the root of C:.
14. Right-click the folder named **rootrepl**.
15. Click **Sharing**.
16. Select **Share this folder**.
17. Keep the share name rootrepl and enter the following in the Comment box: **Share for Dfs root replica**.
18. Click the **OK** button.

Server B is now prepared with a share to use for the root replica. The following steps are for Partner A working at Server A. If you are already logged on, you may skip to Step 7.

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. If your Dfs root does not appear in the console, perform Steps 9 through 12. Otherwise, skip to Step 13.
9. Right-click **Distributed File System** in the Tree pane of the console window.
10. Click **Display an Existing Dfs Root**.
11. In the Dfs root or host server box, type: ***intersales.corp*\salesandmarketing**.
12. Press **Enter**. The domain root appears as \\intersales.corp\salesandmarketing next to a shared server icon in the left pane.
13. Right-click the **root** (\\intersales.corp\SalesandMarketing).
14. Click **New root replica**.
15. In the New Dfs Root Wizard, click the **Browse** button.
16. When the Find Computers dialog box appears, double-click the server name of Server B. The Find Computer dialog box will close and the server name will appear in the Server Name box of the New Dfs Root Wizard in this format: *servername.intersalescorp*. If the server name appears correctly, proceed with the following steps. If it does not, go to the beginning and repeat the steps.
17. Click the **Next** button. The Specify the Dfs Root Share page of the New Dfs Root Wizard appears.
18. In the Use an Existing Share box, use the down arrow, if necessary, to select **rootrepl**.
19. Click the **Finish** button.
20. The Dfs console appears with two root replicas in the contents window.
21. Close the Dfs console.

In this project you have created a root replica, which will enhance availability of Dfs resources.


## Project 6-8 Creating Dfs Link Replicas and Configuring Replication

This project depends on the successful completion of Hands-on Project 6-7. Once again, you will work with a partner. In each pair, the same students and computers will be Partner A, Server A, Partner B, and Server B. Partner B will do the following set of steps in order to create a share to be used as a root replica. If Partner B is already logged on to Server B, Partner B can skip to Step 7.

**Partner B** only

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security Dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears, right-click **My Computer**.
8. Click **Explore**.
9. Double-click **D:**.
10. Click the **File** menu in the menu bar of the window.
11. Point to **New**, and then click **Folder**. This will create a new folder in the root of drive D:.
12. Type **promo** to replace New Folder as the name for the folder, and then press **Enter**. The folder promo should now exist in the root of D:.
13. Right-click the folder named **promo**.
14. Click **Sharing**, and then select **Share this folder**.
15. Keep the share name promo and enter the following in the Comment box: **Share for Dfs link replica**.
16. Click the **OK** button.

Server B is now prepared with a share to use for the link replica. The following steps are for Partner A working at Server A. If you are already logged on, you may skip to Step 7.

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.

4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. When the desktop appears click the **Start** button on the taskbar, point to **Programs**, point to **Administrative Tools**, and then click **Distributed File System**.
8. If your Dfs root does not appear in the console, perform Steps 9 through 13. Otherwise, skip to Step 15.
9. Right-click **Distributed File System** in the Tree pane of the console window.
10. Click **Display an Existing Dfs Root**.
11. In the Dfs root or host server box, type: ***intersales.corp*\salesandmarketing**.
12. Press **Enter**. The domain root appears as \\intersales.corp\salesandmarketing next to a shared server icon in the left pane.
13. If the Dfs links do not appear below \\intersales.corp\salesandmarketing, double-click ***intersales.corp*\salesandmarketing**.
14. Right-click the **Point of Sales** Dfs link.
15. Click **New Replica**. The Add a New Replica Wizard appears.
16. Click the **Browse** button next to the text box.
17. In the Browse for Folder box, double-click **Entire Network**.
18. Double-click **Microsoft Windows Network**.
19. Double-click **Intersales**.
20. Double-click the name of Server B.
21. Click the share **promo**, and then click **OK**.
22. In the Add a New Replica dialog box, ensure that \\serverB\promo is displayed in the text box under Send the user to this shared folder.
23. Under Replication Policy, click to select **Automatic replication**.
24. Click the **OK** button. The Replication Policy dialog box appears.
25. Both Server A (with correct server name) and Server B (with correct server name) will appear in the list of servers, but they will not have replication turned on.
26. Click the server name for Server A.
27. Click the **Set Master** button.
28. Click the server name for Server B.
29. Click the **Enable** button. Replication is now enabled between the replicas for promotion. Server A is the initial master, meaning that the contents of Point of Sales on Server A at this point will be replicated to the promo share on Server B. After this initial replication, they will both be masters, capable of replication to the partner(s).
30. Click the **OK** button to close the Replication Policy menu.


## Optional Hands-on Project Installing the Windows 2000 Administration Tools

The Windows 2000 Administration Tools, including the tools necessary to administer a domain, are only installed automatically on a Windows 2000 server when it is promoted to domain controller. Administrators often need the Administrative Tools on their Windows 2000 Professional or Windows 2000 Member servers. The following are the steps to install the administrative tools from the ADMINPAK.MSI file. If you are already logged on to your server as administrator from the Intersales domain, skip to Step 7.

1. If your server is not powered up, power it up now.
2. Press **Control+Alt+Delete** to display the Security dialog box titled Log On to Windows.
3. In the **User name** box, type **administrator**.
4. In the **Password** box, type **password**. (If this does not work, ask your instructor for the password.)
5. In the **Log on to** box, use the selection arrow to select **INTERSALES**. (This, too, will depend on the classroom configuration.)
6. Press **Return**.
7. Browse to the following folder: c:\winnt\system32.
8. Locate and double-click the file **adminpak.msi**. This will start the Windows Installer, which will load the Windows 2000 Administration Tools Setup Wizard.
9. Click the **Next** button. The installation progress window will appear. If it displays a message concerning file versions, choose whatever option will maintain the newest versions of files on your machine.
10. When the Completing the Windows 2000 Administration Tools Setup Wizard appears, click the **Finish** button.
11. Confirm that the Administration Tools have been installed. Click the **Start** button on the taskbar, point to **Programs**, and then point to **Administrative Tools**.
12. Confirm that you have the following list of tools available from the Administrative Tools menu: Terminal Services Client; Active Directory Domains and Trusts; Active Directory Sites and Services; Active Directory Users and Computers; Certification Authority; Cluster Administrator; Component Services; Computer Management; Connection Manager Administration Kit; Data Sources (ODBC); DHCP; Dfs; DNS; Event Viewer; Internet Authentication Service; Internet Services Manager; Local Security Policy; Performance; QoS Admission Control; Remote Storage; Routing and Remote Access; Services; Telephony; Telnet Terminal Services Licensing; Terminal Services Manager; and WINS.

You have now installed the Windows Administration Tools. Some of these tools will be used in this course.

## CASE PROJECTS


### Enhancing a Dfs Design

In this case project, you will enhance a Dfs design for availability, performance, and security. You are an IT employee of the Intersales company. You have a Dfs design that includes a domain Dfs named \\intersales\salesandmarketing hosted on the two replicas, the servers Danville and Honolulu. Your design also includes two Dfs links hosted on Carmel and Liverpool. Figure 6-12 illustrates this design. Describe how you would further improve this design for availability and performance and what steps you would take to secure the data. If you have access to a drawing program, illustrate some of the changes that you would make to the design.




### Placing Servers to Enhance a Dfs Design

In this case, you will determine the placement of servers to enhance a Dfs design. As part of your Dfs design for Intersales, you need to determine where the servers hosting roots and shares should be located. Intersales has one domain (intersales.corp) and three sites (England, California, and Hawaii). Sales and marketing staff at the headquarters in England need access to the promotions data, while sales and marketing staff in the regional offices in California and Hawaii need access to the point of sales data. Determine where you would place servers, modifying your plan in Case Project 1 if necessary.


### Dfs Command-Line Tools

Your IT staff members often automate administrative tasks by using command line tools in shell scripts (batch files). You have been asked to document the capabilities of the two Dfs command-line tools: DFSCMD.EXE and DFSUTIL.EXE. Document these commands, contrasting their abilities and describing how you would use them.